

BAB II LANDASAN TEORI

2.1 TINJAUAN PUSTAKA

Sebagai bahan perbandingan dan referensi untuk pengembangan penelitian dilakukan kajian pustaka yang diambil dari beberapa jurnal ilmiah tentang penelitian yang relevan dari berbagai sumber di Internet sebagai referensi. Adapun referensi penelitian ini adalah sebagai berikut.

S. Agustin dkk melakukan penelitian mengidentifikasi perkebunan kelapa sawit pada citra satelit pankromatik Ikonos. Ekstraksi fitur yang digunakan domain frekuensi yaitu nilai spektral daya rata-rata radial (RAPSV). Hasil klasifikasi menggunakan machine-learned multilayer perceptron dengan fitur domain spasial menghasilkan akurasi 72,75%, fitur frekuensi 84,75%, dan kombinasi fitur domain spasial dan frekuensi menghasilkan akurasi 86%. Akurasi tertinggi sebesar 89% diperoleh dari kombinasi fitur first-order, fitur lokal, dan RAPSV (Agustin, Ginardi, & Tjandrasa, Identification of Oil Palm Plantation in Ikonos Images using Radially Averaged Power Spectrum Values, 2015).

Penelitian oleh Elva Grecia Anggraini dari Universitas Muhammadiyah Gresik berjudul Sistem Klasifikasi Vegetasi Pada Citra Satelit Lahan Kelapa Sawit Berdasarkan Tekstur Menggunakan Metode Backpropagation. Perbaikan citra dilakukan proses grayscale dan fitur yang digunakan berdasarkan fitur tektur RAPSV. Hasil klasifikasi diberikan berupa citra vegetasi dan citra non vegetasi (Anggraini, 2018).

Pada penelitian yang berjudul “Implementasi Algoritma K-Nearest Neighbor untuk Identifikasi Citra Bunga Anggrek” Hasil yang dari penelitian ini antara lain telah dihasilkan sebuah aplikasi untuk deteksi bunga anggrek berdasarkan kelopak bunga dan didapatkan akurasi sebesar 86,7% (Sari & Wulanningrum, 2021).

Pengenalan Motif Batik Banyuwangi yang dilakukan oleh Hakim dkk, pada eksperimen yang telah dilakukan akurasi terbaik untuk algoritma

KNN sebesar 87,5% dengan parameter K adalah 9 (Hakim, Kristanto, Yusuf, & Afia, 2022).

Klasifikasi Motif Batik Banyuwangi Menggunakan Metode K-Nearest Neighbor (K-NN) Berbasis Android. Referensi yang ada untuk mengidentifikasi tekstur motif batik Banyuwangi menggunakan parameter ekstraksi warna RGB untuk motif batik Banyuwangi. dimana pengujian menggunakan fitur min dan max warna merah, hijau dan biru. Algoritma yang digunakan adalah K-Nearest Neighbors (K-NN). Aplikasi ini mampu mengidentifikasi pola batik khas Banyuwangi dengan akurasi hingga 85% (Harlina & Handayani, 2022).

Penelitian yang dilakukan oleh Zulfrianto Y. Lamasigi Penelitian tersebut bertujuan untuk meningkatkan hasil ekstraksi fitur. Metode yang digunakan adalah *K-Nearest Neighbour*. Dari pengujian diperoleh akurasi tertinggi pada sudut 135° dengan nilai 84,88% untuk $k = 3$, dan terendah pada sudut 0° dengan nilai 41,86% untuk $k = 7$ dan 9 (Lamasigi, 2021).

Klasifikasi Wajah Menggunakan Support Vector Machine (SVM). Pada penelitian tersebut dilakukan dengan sampling dalam variasi posisi pada sudut kemiringan data (-90° , -70° , -45° , -25° , -5°) dan ($+90^\circ$, $+70^\circ$, $+45^\circ$, $+25^\circ$, $+5^\circ$) dengan ukuran citra 640×480 menghasilkan tingkat benar 90% dan salah 10% dari jumlah sampel 200 subjek yang digunakan (Achmad Rizal, Sanjaya Girsang, & Apriyadi Prasetyo, 2019).

Klasifikasi Jenis Buah Cherry Menggunakan Support Vector Machine (SVM) Berdasarkan Tekstur dan Warna Citra. Penelitian ini mengusulkan teknik pengolahan Support Vector Machine (SVM) dalam mengklasifikasi buah cherry. Hasil yang didapat algoritma SVM efisien dalam mengklasifikasi tekstur dan warna buah cherry. Akurasi yang dihasilkan dari aplikasi yang menggunakan algoritma SVM sebesar 100% (Rizky Damanik, Annisa, Islamyati Rafeli, Septiana Liana, & Sandya, 2022).

Klasifikasi 12 Motif Batik Banten Menggunakan *Support Vector Machine*. Penelitian tersebut melakukan pengujian dengan

mengklasifikasikan 12 motif batik Banten menggunakan metode SVM. Hasil klasifikasi 85% menggunakan SVM untuk orde 1 dan 87,2% untuk orde 2 (Wiryadinata, Rofiki, Fahrizal, & Alfanz, 2019).

Penelitian yang dilakukan oleh Mustamin, dkk. Tentang klasifikasi jenis buah alpukat menggunakan metode SVM. Data yang digunakan adalah citra Alpukat total 180 data, yang diklasifikasi menjadi 3 jenis, Alpukat aligator, Alpukat mentega, Alpukat miki. Pada data latih digunakan sebanyak 150 data dan untuk data uji digunakan sebanyak 30 data. Sehingga masing-masing alpukat menggunakan 10 data uji dan diperoleh nilai akurasi alpukat alligator sebanyak 80%, alpukat mentega sebanyak 90% dan alpukat miki sebanyak 50%. Untuk keseluruhan data mencapai nilai akurasi mencapai 73,33% (Hamid, Sudin, & Retna Rumbia, 2022).

Klasifikasi Jenis Jamur Menggunakan metode SVM. Pengujian pada penelitian tersebut menggunakan metode k-fold *Cross Validation* sebanyak empat ($k = 4$). Hasil dari tahapan testing akan dipakai pada metode *confusion matrix* digunakan untuk mendapatkan nilai precision, recall, accuracy. Hasil klasifikasi didapatkan sebesar 23,80% untuk precision, recall sebesar 22,94%, dan untuk accuracy sebesar 82,69% (Yohannes, Udjulawa, & Sariyo, 2022).

2.2 LANDASAN TEORI

2.2.1 Pengolahan Citra

Pengolahan citra adalah istilah umum untuk berbagai teknik yang keberadaannya untuk memanipulasi dan memodifikasi citra dengan berbagai cara (Masse & Ainun, 2018). Pengolahan citra digital (*Digital Image Processing*) merupakan salah satu cabang informatika (ilmu komputer) yang menggunakan komputer dengan teknik pengolahan citra untuk melakukan pengolahan citra dua dimensi.

Pengolahan citra dibagi menjadi dua kelompok yaitu memperbaiki citra sesuai dengan kebutuhan dan mengolah

informasi yang terkandung di dalam citra. Pemrosesan citra biasanya terdiri dari pemrosesan objek citra dengan mengekstraksi informasi penting yang ada di dalamnya. Dalam pengolahan citra diharapkan citra yang telah diproses dapat diambil cirinya (Munir, 2004).

Pemrosesan citra mengubah satu citra menjadi citra lain. Sebuah citra terdiri dari sekumpulan piksel (elemen gambar) dengan koordinat (x,y) dan besaran $f(x,y)$. Koordinat (x,y) merepresentasikan letak/posisi piksel pada citra, sedangkan amplitudo $f(x,y)$ merepresentasikan nilai intensitas warna citra.

2.2.1.1 Citra RGB

Citra berwarna merupakan citra yang memiliki 3 buah kanal warna. Umumnya jenis citra berwarna terbentuk dari komponen merah/ red (R), hijau/ green (G), biru/ blue (B) yang dimodelkan ke dalam ruang warna RGB sehingga citra ini biasa dikenal sebagai citra RGB (Hidayatullah, 2017). RGB adalah nilai positif dalam rentang 0 hingga C_{max} . Nilai C_{max} umumnya adalah 255. Semakin dekat ke nilai 0 maka semakin gelap warna komponen dan semakin dekat ke nilai 255 semakin cerah warna komponen dapat dilihat pada gambar 2.1. Secara matematis, kemungkinan warna untuk piksel i adalah sebagai berikut.

$$C_i = (R_i, G_i, B_i) \quad (2.1)$$



Gambar 2.1. Contoh citra RGB

2.2.1.2 Citra Grayscale

Citra grayscale disebut juga sebagai 8-bit image karena untuk tiap nilai pikselnya memerlukan penyimpanan sebesar 8-bit (Atqiya, Ihsani, Sholahuddin, Dwivany, & Suhandono, 2019). Citra grayscale adalah citra digital dengan hanya satu nilai kernel per piksel, dengan kata lain matriks hanya berisi nilai warna RED=GREEN=BLUE. Nilai ini digunakan untuk menunjukkan intensitas warna, seperti hitam (intensitas 0), abu-abu (intensitas antara 0 dan 255), dan putih (intensitas 255). Pada gambar 2.2 merupakan ilustrasi citra yang mengubah skala keabuan dengan membagi setiap nilai intensitas warna RGB dan kemudian mencari rata-ratanya. Citra yang hanya memiliki 1 buah kanal sehingga yang di tampilkan hanya nilai intensitas atau di kenal juga dengan istilah derajat keabuan (Triprasetyo, Pamungkas, & Wulanningrum, 2018).

$$Gray = rgb2gray (image) \quad (2.2)$$

atau

$$Gray = (0,2989)*R + (0,587)*G + (0,114)*B \quad (2.3)$$



Gambar 2.2. Contoh citra grayscale

2.2.2 Preprocessing

Pada tahap ini proses peningkatan kualitas citra, dan tujuannya untuk meningkatkan tingkat keberhasilan pada tahapan pengolahan citra digital berikutnya, seperti *grayscale*, *low-pass filtering*, *resizing*, *sharpening*, *histogram equalization* dan *thresholding*. *Preprocessing* diperuntukan menghilangkan bagian-bagian yang tidak diperlukan pada gambar.

2.2.2.1 Resizing

Resizing adalah proses mengubah ukuran matriks citra menjadi ukuran matriks tertentu. Gambar masukan awal akan diubah ukurannya dari ukuran asli ke ukuran gambar yang diinginkan. Tahap *resizing* dalam normalisasi adalah mengubah ukuran gambar training dan test. Mengubah ukuran gambar dapat menghasilkan gambar yang lebih besar atau lebih kecil dari gambar asli.

2.2.2.2 Contrast Limited Adaptive Histogram Equalization (CLAHE)

Contrast Limited Adaptive Histogram Equalization (CLAHE) merupakan enhancement pengembangan dari *Histogram Equalization*. Dalam metode ini yang dikembangkan secara independen adalah pemetaan

peningkatan kontras yang diterapkan untuk hal tertentu (Reza, 2004) .

CLAHE merupakan teknik perbaikan kontras citra dengan meningkatkan kontras lokal citra. Meningkatkan warna dan tampilan objek yang tidak jelas pada citra. algoritma CLAHE membagi citra ke dalam area-area yang lebih kecil dan menerapkan HE untuk masing-masing area tersebut.

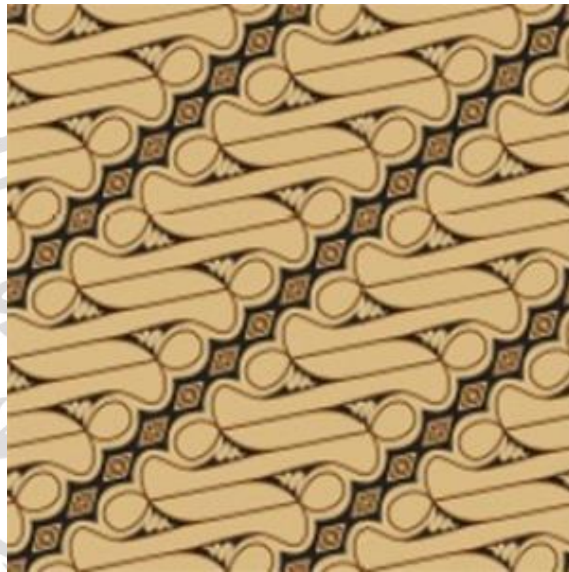
2.2.3 Batik

Batik berasal dari Bahasa Jawa yaitu “amba” yang artinya tulis dan “nitik” yang berarti titik. Maksud dari gabungan kedua kata tersebut adalah menulis dengan lilin. Proses pembuatan batik diatas kain menggunakan canting yg ujungnya berukuran kecil memberikan kesan “orang sedang menulis titik-titik”. Di samping itu batik memiliki pengertian yang berhubungan dalam membuat titik atau meneteskan lilin atau malam pada kain mori (Trixie, 2020).

Batik adalah suatu bentuk hasil cipta karya seni yang indah, yang diekspresikan pada pola kain untuk pakaian, sarung, kain panjang, dan kain dekoratif lainnya. Secara harfiah batik dijelaskan sebagai kain bergambar yang dibuat secara khusus dengan menuliskan atau menorehkan malam (lilin) pada kain dan kemudian pengolahannya diproses dengan cara tertentu. Teknik pembuatan batik dikerjakan dengan cara cap, printing (sablon), kain tekstil bercorak batik, batik dengan komputer, serta batik tulis (Triprasetyo, Pamungkas, & Wulanningrum, 2018).

Membatik pada dasarnya sama dengan melukis di atas sehelai kain putih (kain polos). Sebagai alat melukisnya dipakai canting, dan sebagai bahan melukisnya dipakai cairan malam atau lilin. Setelah kain dibatik diberi warna, kemudian lilin dihilangkan,

maka bagian yang tertutup lilin atau malam akan tetap putih, tidak menyerap warna. Ini disebabkan karena lilin berfungsi sebagai perintang warna. Proses inilah akan menghasilkan kain batik (Aziz & Maimunah, 2017).



Gambar 2.3. Contoh gambar pola motif batik

2.2.4 Ekstraksi Fitur

Proses ekstraksi fitur adalah proses pencarian ciri agar sistem dapat mengenali karakteristik unik dari masing-masing motif batik yang diwujudkan dalam metode pengolahan citra (pengenalan motif). Tujuan utama ekstraksi fitur adalah untuk memperoleh informasi yang paling relevan dari data asli dan merepresentasikan informasi tersebut dalam ruang dimensi yang lebih rendah (Usman, 2005).

2.2.4.1 Fast Fourier Transform (FFT)

Fast Fourier Transform (FFT) adalah teknik perhitungan operasi matematika yang digunakan untuk mentransformasi sinyal analog menjadi sinyal digital

berbasis frekuensi. *Fast Fourier Transform* (FFT) membagi sebuah sinyal menjadi frekuensi yang berbeda-beda dalam fungsi eksponensial yang kompleks (Kusuma, 2021). *Fast Fourier Transform* (FFT) adalah Transformasi Fourier yang dikembangkan dari algoritma discrete transform fourier (DFT) (Abdillah, 2017). FFT adalah algoritma untuk menghitung Discrete Fourier Transform (DFT) dengan cepat dan efisien (Radiana, 2008).

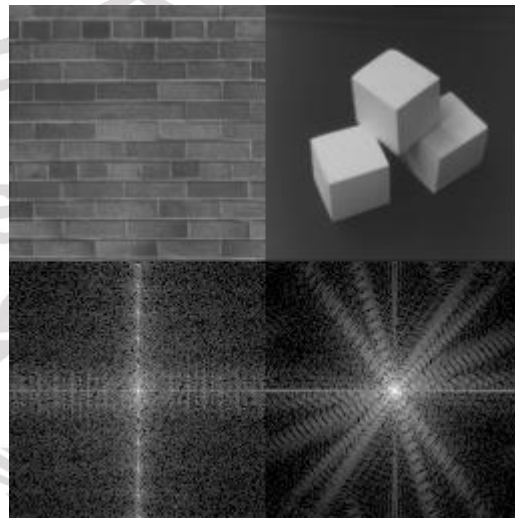
Discrete Fourier Transform (DFT) adalah teknik untuk mengubah bentuk sinyal dari domain waktu ke domain frekuensi. Namun DFT memiliki kelemahan dalam hal efisiensi waktu untuk menghitung isi frekuensi, sehingga digunakan algoritma *Fast Fourier Transform* (FFT) yang lebih efisien dalam mengubah sinyal dari representasi domain waktu ke domain frekuensi (Jacobus, Ruindungan, & Litouw, 2019).

Fast Fourier Transform (FFT) adalah algoritma Transformasi Fourier Diskrit yang mengurangi jumlah perhitungan yang diperlukan oleh N dari $2N^2$ menjadi $2N \log N$, dimana \log adalah logaritma berbasis 2. Algoritma Fourier Transform yang nyata dan efisien memberikan peningkatan lebih lanjut dalam kecepatan untuk sekitar dua factor. Base-4 dan Base-8 FFT menggunakan kode yang dioptimalkan, dan dapat 20-30% lebih cepat dari Base-2FFT (Abdillah, 2017).

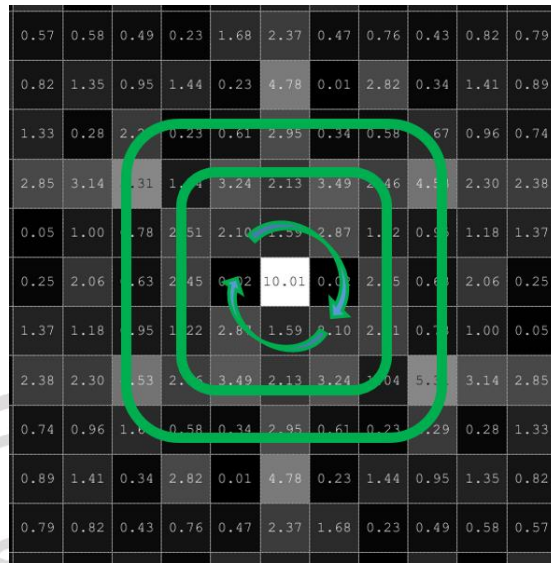
2.2.4.2 Radially Averaged Power Spectrum Value (RAPSV)

RAPSV digunakan untuk menghitung spektrum daya rata-rata radial dari hasil transformasi fourier. Hasil dari transformasi Fourier dapat digambarkan sebagaimana

sebuah citra. Untuk keperluan visualisasi spektrum, dilakukan penggeseran dari hasil transformasi Fourier. Beberapa contoh spektrum Fourier yang telah dilakukan penggeseran disajikan pada gambar 2.4 (Agustin, Segmentasi Lahan Perkebunan Kelapa Sawit Berdasarkan Usia Tanam Pada Citra Satelit, 2021).



Gambar 2.4. Spektrum fourier yang terbentuk dari citra
(Agustin, *Segmentasi Lahan Perkebunan Kelapa Sawit Berdasarkan Usia Tanam Pada Citra Satelit, 2021*)



Gambar 2.5. Ilustrasi penghitungan fitur RAPS_{SV} (Agustin, *Segmentasi Lahan Perkebunan Kelapa Sawit Berdasarkan Usia Tanam Pada Citra Satelit*, 2021)

Ilustrasi penghitungan fitur RAPS_{SV} ditunjukkan pada gambar 2.5.

(Agustin, *Segmentasi Lahan Perkebunan Kelapa Sawit Berdasarkan Usia Tanam Pada Citra Satelit*, 2021).

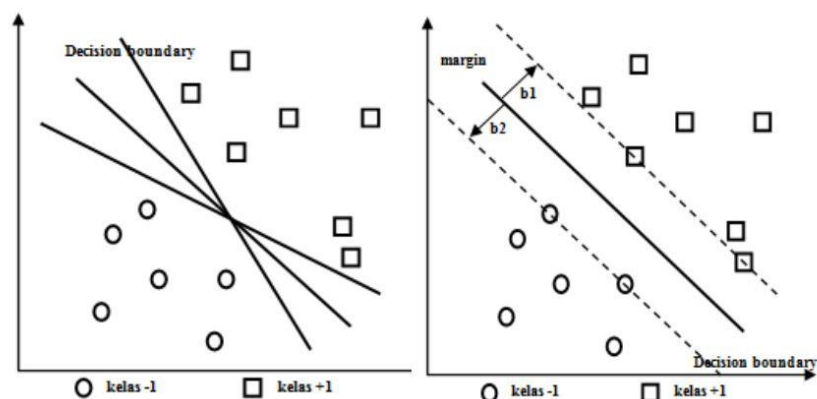
2.2.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) pertama kali diperkenalkan oleh Vapnik pada tahun 1992 bersama rekannya Bernhard Boser dan Isabelle Guyon (Widodo, Handayanto, & Herlawati, 2013). Saat pertama kali diperkenalkan, SVM hanya terbatas untuk mengklasifikasikan data menjadi dua kelas (Sembiring, 2007). Metode ini bekerja dengan cara mendefinisikan batas antara dua kelas dengan jarak maksimal dari data yang terdekat (Clarke, Fokoue, & Zhang, 2009).

Pada prinsipnya, SVM bekerja secara linier dan dikembangkan untuk masalah nonlinier. Dengan menggunakan metode kernel trick, sebuah *hyperplane* dicari dengan

mentransformasikan dataset ke dalam ruang vektor yang berdimensi lebih tinggi (*feature space*), kemudian dilakukan proses klasifikasi pada *feature space* tersebut. Penentuan fungsi kernel yang digunakan akan sangat mempengaruhi hasil prediksi.

Support Vector Machine (SVM) merupakan metode *machine learning* yang bekerja dengan prinsip *Structural Risk Minimization* (SRM) untuk menemukan *hyperplane* yang memisahkan dua kelas dalam inputan. Metode tersebut menggunakan bentuk fungsi linier pada ruang fitur berdimensi tinggi dengan menerapkan bias pembelajaran yang diturunkan dari teori pembelajaran statistik. Tingkat akurasi model yang akan dihasilkan oleh proses transformasi menggunakan SVM sangat bergantung pada fungsi kernel dan parameter yang digunakan. Ide dasar dari metode SVM adalah untuk memaksimalkan batas *hyperplane*, seperti yang ditunjukkan pada gambar 2.6.



Gambar 2.6. Batas keputusan yang mungkin untuk dataset (Prasetyo, 2014)

2.2.6 K-Nearest Neighbour (KNN)

K-Nearest Neighbour adalah metode parameter klasifikasi sederhana. Klasifikasi ini membutuhkan metrik dan bilangan interger positif. Tujuan dari algoritma ini adalah untuk mengklasifikasikan objek baru berdasarkan nilai atribut dan data

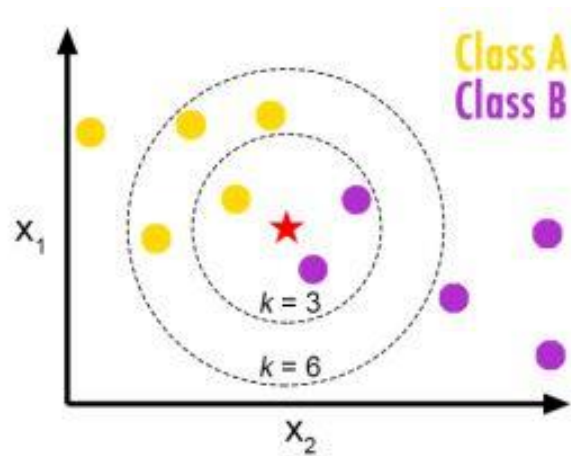
pelatihan. *K-Nearest Neighbour* (KNN) adalah metode untuk mengklasifikasikan objek berdasarkan data yang dipelajari (*neighbour*) yang paling dekat jaraknya dengan objek tersebut. Tetangga dekat atau jauh biasanya dihitung berdasarkan jarak Euclidean. Metode ini menggunakan algoritma supervised, dimana hasil sampel uji baru diklasifikasikan menurut mayoritas kelas di KNN. Algoritma metode KNN sangat sederhana, KNN ditentukan berdasarkan jarak terpendek dari query instance ke training sample. Rumus perhitungan mencari jarak Euclid sebagai berikut

$$d = \sum_{i=1}^n (x_i - y_i)^2 \quad (2.5)$$

Keterangan :

- x_i = data uji
- y_i = data latih
- i = variabel data
- d = jarak

Metode KNN dibagi menjadi dua fase, yaitu pembelajaran (training) dan klasifikasi atau pengujian (testing). Pada fase pelatihan, algoritma ini hanya menyimpan vektor-vektor fitur dan mengklasifikasi data pelatihan. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk data yang akan diuji (yang klasifikasinya tidak diketahui). Jarak dari vektor baru ini ke semua vektor data pelatihan dihitung, dan diambil k neighbor yang paling terdekat (Baharuddin, Hasanuddin, & Huzain, 2019). Sebuah titik akan diprediksi jenisnya berdasarkan klasifikasi terbanyak dari *neighbor* disekitarnya, ilustrasi dapat dilihat pada gambar 2.7.



Gambar 2.7. Ilustrasi penggunaan nilai k (Baharuddin, Hasanuddin, & Huzain, 2019)

Berikut langkah-langkah untuk menghitung algoritma KNN.

1. Menentukan nilai parameter k (jumlah tetangga yang paling dekat).
2. Menghitung jarak Euclid antara data baru dan semua data pelatihan.
3. Urutkan jarak menurut jarak Euclid terkecil ke- k .
4. Kumpulkan kelas klasifikasi tetangga terdekat sebagai nilai prediksi baru.